Automatic Car Parking: A Reinforcement Learning Approach

Darío Maravall[1], Miguel Ángel Patricio[2], and Javier de Lope[1]

[1] Department of Artificial Intelligence
Faculty of Computer Science
Universidad Politécnica de Madrid
Campus de Montegancedo, 28660 Madrid, Spain
{dmaravall,jdlope}@dia.fi.upm.es
[2] Departamento de Informática
Universidad Carlos III de Madrid
mpatrici@inf.uc3m.es

Abstract. The automatic parking of a car-like robot is the problem considered in this paper to evaluate the role played by formal representations and models in neural-based controllers. First, a model-free control scheme is introduced. The respective control actions are sensory-based and consist of a dynamic, neural-based process in which the neurocontroller optimizes *ad hoc* performance functions. Afterwards, a model-based neurocontroller that builds without supervised a formal representation of its interaction with the environment is proposed. The resulting model is eventually utilized to generate the control actions. Simulated experimentation has shown that there is an improvement in robot behavior when a model is used, at the cost of higher complexity and computational load.

1 Introduction

In this paper we investigate the role played by representations and models of the environment in artificial neurocontrollers. We have chosen the automatic parking of car-like robots to illustrate this discussion. Previous work on this subject has been based either on hard computing techniques [1]-[3] –i.e., analytical methods from applied mathematics and control theory- or soft computing techniques, like fuzzy logic and artificial neural networks (ANN), [4]-[6]. A common feature of both hard computing and soft computing methods for automatic parking maneuvers is that they are based on *a priori* known models of the environment, as well as known car kinematics and dynamics. Sometimes, particularly for ANN-based methods, the aprioristic models of the environment are replaced by supervised trajectories that allow the vehicle controller to learn the proper maneuvers.

We depart from the usual model-based and supervised approaches to propose a model-free and unsupervised technique to solve the automatic parking problem. The paper is organized as follows. First, we succinctly present the problem at hand and introduce the general description of our method, which is based

J. Mira and J.R.Álvarez (Eds.): IWANN 2003, LNCS 2686, pp. 214-221, 2003.

on sensory-driven reinforcement control. Afterwards, we introduce the ANN implementation of the proposed method and present experimental results. Finally, we discuss in detail the role played by *a posteriori* formal representations or models of the interaction between the vehicle and the environment, emphasizing the differences between such learned models and the usual *a priori* models or equivalent supervised techniques –i.e., induced trajectories-.

2 Automatic Parking with Reinforcement Control

Commercial cars can be modelled as dynamic systems with, basically, two control inputs: speed, v, and steering angle, ϕ. By admitting the usual hypothesis of the car being on a plane, the state variables are (x, y, θ), as illustrated in Figure 1, in which we have considered the two typical kinematics of commercial cars; i.e., rear traction and front traction. The respective dynamical equations are

$$(a)\ \ \dot{x} = |v| \cos\theta\,; \quad \dot{y} = |v| \sin\theta; \quad \dot{\theta} = \frac{|v|}{L}\tan\phi \tag{1}$$

$$(b)\ \ \dot{x} = |v| \cos(\theta + \phi)\,; \quad \dot{y} = |v| \sin(\theta + \phi); \quad \dot{\theta} = \frac{|v|}{L}\sin\phi$$

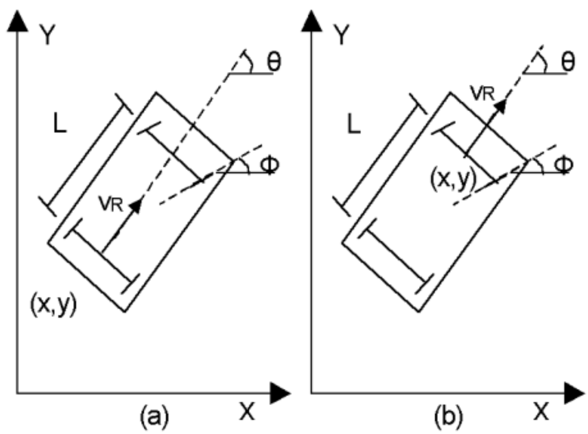

Fig. 1. Two typical kinematics of commercial cars: (a) rear traction and front steering and (b) front traction and front steering.

Let us suppose that our car-like robot is controlled only by the steering angle –i.e., it moves at a constant speed-. The automatic car parking task using conventional control theory [7] is based on the typical feedback loop. In this case, $\theta_d(t)$ is the desired orientation of the car, which is compared with its current direction and, as a result of the error, a control law generates the steering angle that eventually drives $\theta(t)$ towards $\theta_d(t)$. Another two similar control loops for the remaining state variables, x and y, are needed for complete lateral control of the vehicle. As mentioned before, the main difficulty in applying control theory to automatic car parking is the computation of the desired control trajectories $\theta_d(t)$, $x_d(t)$ and $y_d(t)$, which are always pre-computed and injected into the car-like robot in real-time. When using soft computing techniques, either fuzzy logic or ANN, the rigid control loops take another shape.

In fuzzy logic-based automatic parking of a car, [4], the control laws are converted into linguistic rules and the environment's responses are also converted into linguistic variables. As is well known, the design process and, particularly, the tuning of membership functions is very cumbersome and tricky. Another problem is the possible combinatorial explosion of the fuzzy control rules, [6].

When applying ANN techniques, [4] and [5], the control laws of the car-like robot are learnt by sensory-based experience and, particularly, by following a teacher that provides the correct actions for each specific situation. Although the ANN approach simplifies the design process substantially, the fact is that it calls for an exhaustive battery of training examples and, as is very well known, the system can get into serious trouble in the operating stage, if it comes up against situations for which it has not been trained.

We have introduced elsewhere, [8] and [9], a method based on the perception-planning-action cycle that is able to solve the navigation of robotic mechanisms without explicit knowledge of their kinematics and without using formal representations or models of their interaction with unknown environments. This method is bio-inspired at the functional level and its particularization to the automatic parking problem is shown in Figure 2.

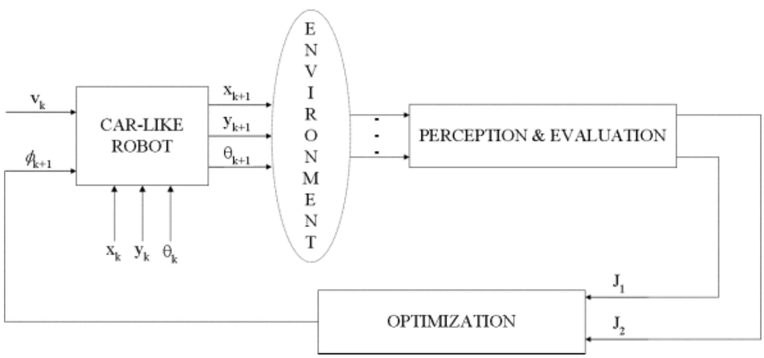

Fig. 2. Block-diagram of the proposed system for automatic parking maneuvers.

In this particular case, there are two performance functions J_1 and J_2, which represent (1) the objective of the car-like robot being parked in the final position (x_d ,y_d) and (2) pointing in a desired orientation $\theta_d(t)$. Therefore,

$$
\begin{aligned}
J_1 &= \frac{1}{2}\left[(x-x_d)^2 + (y-y_d)^2\right] \\
J_2 &= \frac{1}{2}\left(\theta-\theta_d\right)^2
\end{aligned}
\tag{2}
$$

As mentioned above, let us suppose that the robot is driven by a negative constant speed –ie., it is maneuvering to reverse- so that there is a single control variable ϕ. Obviously, the control law is aimed at minimizing both performance indices:

$$
\dot{\phi}(t) = -\mu_1 \frac{\partial J_1}{\partial \phi} - \mu_2 \frac{\partial J_2}{\partial \phi} \tag{3}
$$

where μ_1 and μ_2 weight the relative importance of each objective. The discrete-time version of this control law is

$$
\phi_{k+1} = \phi_k - \mu_1 \left.\frac{\partial J_1}{\partial \phi}\right|_{\phi(k)} - \mu_2 \left.\frac{\partial J_2}{\partial \phi}\right|_{\phi(k)} \tag{4}
$$

In other words, the car steering angle is computed as an increment/decrement step of the current steering angle, whose sign and module depend on the gradients of both performance functions.

3 Model-Free Behavior

By computing the empirically estimated gradient of the sensory-based performance indices $J_1(\phi)$ and $J_2(\phi)$, our robot is acting as a model-free agent, immerse in a dynamic environment. The agent decisions –i.e., its control actions- exclusively rely on the direct and instantaneous information gathered from the environment by its sensors :

$$
\left.\frac{\partial \hat{J}_1}{\partial \phi}\right|_{\phi(k)} = \frac{J_1(k) - J_1(k-1)}{\phi(k) - \phi(k-1)}\,; \quad \left.\frac{\partial \hat{J}_2}{\partial \phi}\right|_{\phi(k)} = \frac{J_2(k) - J_2(k-1)}{\phi(k) - \phi(k-1)} \tag{5}
$$

This sensory-based behavior can be interpreted as purely reactive, in the sense that perception and action are directly coupled, without any type of intermediate processing. However, this simple and model-free behavior is extremely powerful, as we find by looking at the experimental results shown in Figure 3, in which the car-like robot has successfully performed several parking maneuvers.

The secret of the excellent performance achieved by the robot lies in two elements: (1) the objective functions evaluating its performance, which are in

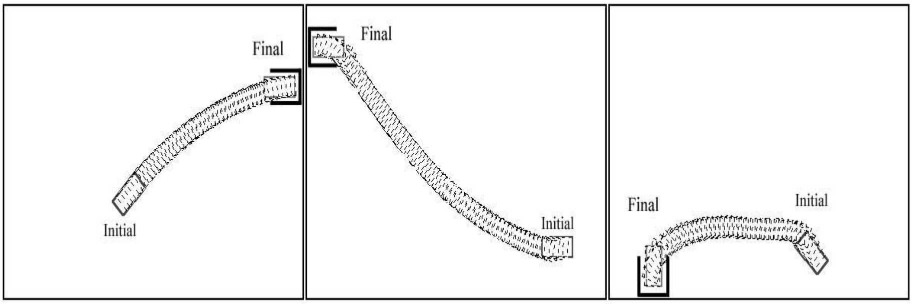

Fig. 3. Several parking maneuvers executed by the robot using a simple, model-free behavior.

turn based on sensory information, and (2) the action strategy based on the optimization of the performance functions. Furnished with these two elements, the car-like robot is able to interact with its environment without using a formal representation or model of its interaction with the environment, and without making explicit use of its own kinematics –i.e., the transformation between its internal coordinates and the environment coordinates-.

4 Model-Based Behavior

Let us now investigate the use of formal representations of the vehicle-environment ensemble. We emphasize that we are talking about modelling the joint dynamics of the agent and its particular environment, rather than modelling each part individually. Then, we are going to evaluate robot performance in parking when its behavior is based on a formal representation or model of its interaction with the environment. Figure 4 represents the modelling process undertaken by the car-like robot by means of an ANN. Note that the model built by the ANN represents the relationship between the vehicle actions –i.e., the steering angles- and the responses of the environment, as quantified by performance indices. This ANN has been implemented via a multilayer perceptron with backpropagation of the model error.

It should be noted that we have explicitly introduced a second ANN to produce the control actions. This ANN, as shown in Figure 4, is based on the same idea of utilizing the empirically estimated gradients of the indices and it has been implemented by means of a simple perceptron. Then, we can conclude that the agent's formal representation of its interaction with the environment is not used for control purposes, as the steering angle is exclusively based on the gradient of the performance functions obtained from the sensor readings. However, once an accurate model of the vehicle-environment ensemble has been built by the respective ANN, then the agent actions can also be generated by the model. This kind of model-based control is known as indirect control, as

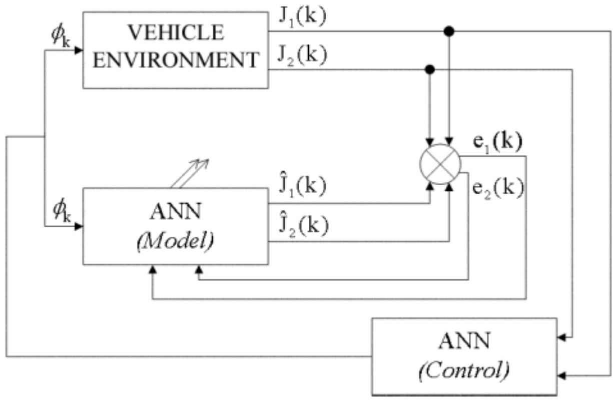

Fig. 4. Conceptual block-diagram of the modelling process using an ANN. Note that the control actions are generated by another ANN minimizing the performance functions.

it is obtained from a model of the plant under control [10]. Figure 5 shows the block-diagram of an hybrid control architecture, in which we have simultaneously considered both type of control actions: (1) model-free or direct control and (2) model-based or indirect control. Note also the ANN in charge of the tricky task of properly combining both control actions.

In both direct and indirect control, the respective control actions ϕ_k^e and ϕ_k^m, respectively, depend on the sensory-based gradients, either as measured by the sensors or as estimated by an ANN. Due to this sensory dependency, the modules of the respective control outputs tend to be disproportionated to the internal dynamics of the vehicle: sometimes the sensory-based gradients are too strong or, inversely, sometimes they are too small, as far as the vehicle's natural dynamics is concerned. To solve this problem of scale, we have devised a look-up table mapping the module of the sensory-based gradients to the natural dynamics of the car-like robot, for both the model-free and the model-based behaviors.

Figure 6 presents the results obtained with the model-based or indirect control actions, for the same situations considered for the direct actions illustrated in Figure 3. We find that the use of a formal representation of the robot-environment couple has appreciably improved the smoothness of the vehicle trajectories. As a conclusion, at least for automatic parking maneuvers, the use of models improves the robot performance. We must add, however, that this improvement is accomplished at the cost of a more complex design process. Tuning the ANN in charge of the modelling process and tuning the ANN combining the direct and the indirect control actions is a tricky and cumbersome process. As a general conclusion, the use of formal representations involves, as in other fields, a subtle trade-off between efficiency and complexity.

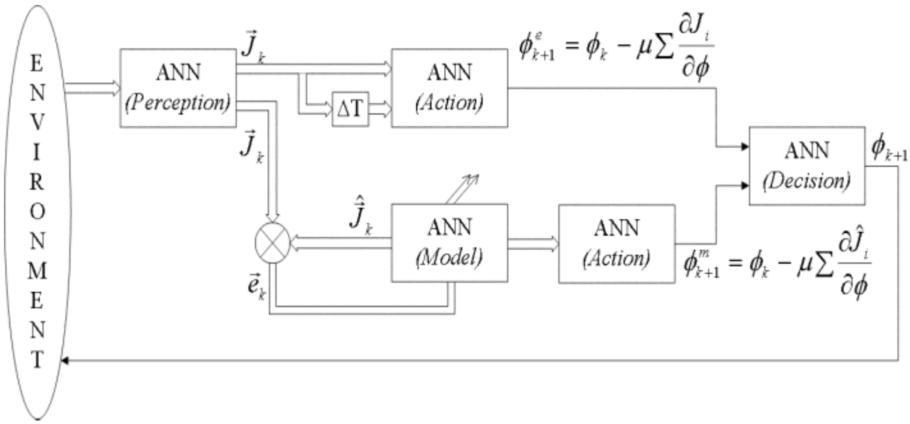

Fig. 5. Block-diagram of the combined direct and indirect control actions. Note the vectors representing more than just one signal or function.

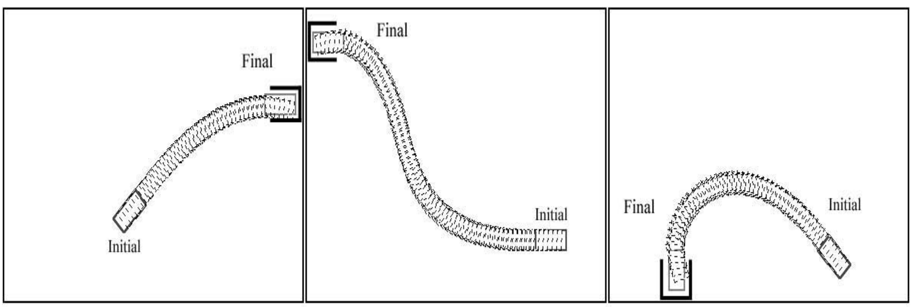

Fig. 6. Experimental results obtained with the model-based or indirect control.

5 Conclusions

Automatic car parking maneuvers is the problem considered in this paper to evaluate the role played by and the pros and cons of formal representations of the interaction of a physical agent with its environment. First, we discussed our previous work on reinforcement control without the use of models of the environment and, then, we introduced a model-free and ANN-based control scheme for performing parking maneuvers. Afterwards, we introduced a model-based or indirect control scheme, also implemented using ANN, for the same task. Simulated experimentation of both direct and indirect reinforcement control has shown that model-based control actions are more efficient –in particular, they perform the parking maneuvers smoother- than model-free or direct actions, at the cost, however, of increased design complexity and computational burden.

Acknowledgements

This work has been partially supported by the Spanish Ministry of Science and Technology, project DPI2002-04064-CO5-05.

References

1. Laumond, J.-P., Jacobs, P.E., Taix, M., Murray, R.M. (1996) A motion planner for non-holonomic mobile robots. IEEE Trans. On Robotics and Automation, 10(5), 577–593.
2. Paromtchik, I.E., Laugier, C. (1996) Motion generation and control for parking an autonomous vehicle. Proc. IEEE Int. Conference on Robotics and Automation, Minneapolis, MN, 3117-3122.
3. Laugier, C., Fraichard, Th., Garnier, Ph., Paromtchik, I.E., Scheuer, A. (1999) Sensor-based control architecture for a car-like vehicle. Autonomous Robots 6, 165-185.
4. Kong, S.G., Kosko, B. (1992) "Comparison of fuzzy and neural track backer-upper control systems". In B. Kosko (ed). Neural Networks and Fuzzy Systems. Prentice-Hall. Englewood Cliffs, NJ, 339-361.
5. Gu, D., Hu, H. (2002) Neural predictive control for a car-like mobile robot. Robotics and Autonomous Systems 39, 73-86.
6. Hitchings, M., Vlacic, L., Kecman, V. (2001) "Fuzzy control". In L. Vlacic, M. Parent, F. Harashima (eds). Intelligent Vehicle Technologies. Butterworth&Heinemann, Oxford, 289-331.
7. Canudas de Wit, C. (1998) "Trends in mobile robots and vehicle control". In B. Siciliano, K.P. Valavanis (eds). Control Problems in Robotics and Automation. LNCIS 230, Springer, Berlin, 151-176.
8. Maravall, D., de Lope, J. (2002) "A reinforcement learning method for dynamic obstacle avoidance in robotic mechanisms". In D. Ruan, P. D´hondt, E.E. Kerre (eds). Computational Intelligent Systems. World Scientific, Singapore, 485-494.
9. Maravall, D., de Lope, J. (2003) "A bio-inspired robotic mechanism for autonomous locomotion in unconventional environments". In C. Zhou, D. Maravall, D. Ruan (eds). Autonomous Robotic Systems: Soft Computing and Hard Computing Methodologies and Applications. Physica-Verlag, Springer, Heidelberg, 263-292.
10. Zhou, C., Meng, Q. (2003) Dynamic balance of a biped robot using fuzzy reinforcement learning agents. Fuzzy Sets and Systems, 134(1), 169-187.